10 Rec'd PCT/PTO 16 JUN 2000

09/582047

METHOD AND APPARATUS FOR SWITCHING DATA

Technical field of the invention

The present invention refers to a method and an apparatus for for switching data in a circuit switched synchronous time division multiplexed network from a first bitstream, being received at a first port of a switch, to a second and a third bitstream, being transmitted from at a second and a third port, respectively, of said switch.

Technical Background and Prior Art

Today, new types of circuit-switched communication networks are being developed for the transfer of information using synchronous or isochronous, time division multiplexed bitstreams, wherein a bitstream is divided into cycles, each cycle in turn being divided into time slots.

An example of such a network is described in "The DTM Gigabit Network", Christer Bohm, Per Lindgren, Lars Ramfelt, and Peter Sjödin, Journal of High Speed Networks, 3(2):109-126, 1994, and in "Multi-gigabit networking based on DTM", Lars Gauffin, Lars Håkansson, and Björn Pehrson, Computer networks and ISDN Systems, 24(2), 119-139, April 1992.

In such a network, so called switches, each connected to one or more bitstreams, or fibers, are used to switch time slot data between different bitstreams. For example, if a channel is defined between a first and a second apparatus attached to a first and a second, respectively, bitstream carrying fiber, said channel comprising a first set of time slots on a bitstream propagating on the first fiber and a second set of slots on a bitstream propagating on the second fiber, a switch is then used to transfer or copy time slot data from time slots of the first set of slots to time slots of the second set of slots.

According to prior art, switches in synchronous time division multiplexed networks use a control memory that

maps each incoming slot number to the outgoing slot number. Such mapping may involve both a mapping in the time domain, i.e. control of the order in which time slot data are written into each bitstream, and a mapping in the space domain, i.e. controlling which time slot data goes to which bitstream. For example, so called time-space-time (TST) switches are described in "Data and Computer Communications", 4th ed., by Williams Stallings, Macmillan Publishing Company. However, prior art switches all show limitation as to the possibilities of switching time slots in space and time. Also, prior art switches show limitations as to switching speed and capacity, especially in the context of switching data from one bitstream to two or more bitstreams, i.e. in the context of space multicasting or broadcasting.

Objects of the invention

An object of the invention is therefore to provide a switch which provides greater freedom as to the possibilities of switching time slots in space and time, at the same time increasing switching speed and capacity.

Summary of the invention

The above mentioned and other objects are achieved by the invention as defined in the accompanying claims.

The invention thus provides a method and an apparatus for for switching data in a circuit switched synchronous time division multiplexed network from a first bitstream, being received at a first port of a switch, to a second and a third bitstream, being transmitted from a second and a third port, respectively, of said switch. According to the invention, data read from a time slot position of said first bitstream at said first port is associated with a channel identifier that identifies a channel that said time slot position forms part of. The channel identifier is then used at said second port for mapping said data into a time slot position of said

second bitstream as well as at said third port for mapping said data into a time slot position of said third bitstream.

The invention is hence based upon the idea of transferring time slot data, belonging to a specific channel, read from the first bitstream at an input port of said switch, to two or more output ports of the switch without providing any direct specification of a time slot in the second bitstream into which the data shall be written. i.e. not having to specify, at channel set-up, a strictly unambiguous slot-to-slot mapping through the switch. Instead, resolution is achieved in connection with the output port by means of an identification of the channel. This provides the advantage that there is no need for any strict timing requirements between the individual time slots of a channel at the input port and the individual time slot of the corresponding channel at the output port, but the phase of the channel on one of the bitstreams may be arbitrary shifted in relation to the phase of the corresponding channel on another bitstream.

Furthermore, the same channel identifier is used at both of said two or more output ports. In other words, when performing space multicasting or broadcasting, both of said two or more output ports are arranged to recognize and map data based upon the same channel identifier. Consequently, the input port does not have to generate different channel identifiers for different output ports for identifying one single channel, thereby simplifying the design of the input port.

Even though the use of channel identifiers as such may have been mentioned in prior art, no prior art is found wherein a channel identifier is used for mapping at at more than one output port in order to provide space multicasting or broadcasting in the context of the invention.

Mapping of data to time slots based upon the identification of a channel preferably comprises mapping said

data itto the next available time slot of said channel on said second and said third bitstream. The advantage here lies in that there is no need to wait for a certain slot of the channel in the second bitstream and the third bitstream. Instead, data is directly mapped to the first time slot available to the channel that has not yet been filled with switched data on the respective output bitstream.

Also, mapping of data from time slots of a channel from said first bitstream to said second bitstream and to said third bitstream is preferably done in maintained mutual order. This is of course required in many applications, and when used in such a context the switch must be able to meet this requirement.

According to the invention, data may be transferred from the input port to the output ports using allocated time slots of an internal bitstream, said bitstream typically being divided into cycles which in turn are divided into time slots. Said internal bitstream is optionally shared by several input and output ports.

Such an internal bitstream simplifies the internal handling of the time slot data in the node. For example, when data is to be multicasted from one bitstream to several other bitstreams, there is no need to make copies of the data to be switched since it will be transferred to all parts of the node having connections with the other bitstreams. Also, this put less requirement on the switch internal operations to be synchronized to the external bitstreams. In fact, it enables the switch to switch data asynchronously within the node. However, a synchronous operation is also preferred, depending on the actual application.

The channel identification is preferably achieved through associating said read time slot data with a channel identifier, which may be performed in many ways, as suggested by the dependent claims of the invention.

When for example connecting, or tagging, read data

with a channel identifying header, these are preferably transferred within said node using time slots of an internal bitstream of the kind described above. Of course, time slot data and an associated channel identifier may also be transferred within the node using private connections, multiplexors, or other transferring means.

Preferrably, mapping of data to time slot of output bitstreams is realized using channel based FIFO buffer, or by using a round robin scheme, examples thereof being described below with reference to Fig. 6a, 6b, and 7.

The invention is especially advantageous in multichannel networks wherein channels are of arbitrary size and wherein channel size may vary dynamically, such as a DTM network of the kind mentioned above.

The present application corresponds to one three swedish patent applications, SE 9704738-5, SE 9704739-3, and SE 9704740-1, which were filed on the same day and which refer to related inventive ideas, the descriptions thereof hereby being incorporated by reference.

The above mentioned and other aspects and features of the invention will be more fully understood from the following description, with reference to the accompanying drawings, of exemplifying embodiments thereof.

Brief description of the drawings

Exemplifying embodiments of the invention will be described below with reference to the accompanying drawings, wherein:

Fig. 1 schematically shows a cahnnel identifying table used in exemplifying embodiments of the invention;

Figs. 2-5 schematically show respective exemplifying embodiments of the invention; and

Figs. 6a, 6b, and 7 show embodiments of mapping means included in the switch nodes shown in Figs. 2-5.

Detailed description of preferred embodiments

Fig. 1 shows a channel identifying table 240 used in the embodiment to be described below with reference to Fig. 2. The channel identifying table in Fig. 1 has one entry for each time slot position of a frame of the input bitstream. For each position, i.e. at each entry, the table provides information designating the channel that the time slot position forms a part of.

A switch according to a first embodiment of the invention will now be described with reference to Fig. 2. In Fig. 2, a switch node 210 is shown having a first port 225 for receiving a first first bitstream 215, a second port 230 for transmitting a second bitstream 220, and a third port 231 for transmitting a third bitstream 221.

The port 225 includes a bit clock, a slot counter and a frame clock (neither of which is shown in the figure). The bit clock is synchronized to the bitrate of the bitstream 215 and is used as input to the slot counter. The slot counter counts the number of slots received from bitstream 215 and is cyclically restarted by the frame clock at the start of each new cycle.

The switch is by means of a controller 235 configured to switch data from a specific set of time slots in frames transferred by bitstream 215 to a specific set of time slots in frames transferred by bitstream 220 and a specific set of time slots in frames transferred by bitstream 221. When establishing a channel through the switch 210, the controller 235 is informed of which time slots of bitstream 215, which time slots of bitstream 220, and which time slot of bitstream 221 that are allocated to the channel.

The controller 235 is connected, as schematically illustrated by dashed lines in Fig. 2, to a channel identifying table 240 of the kind described above with reference to Fig. 1 and to mapping means 250, 251, which will be described more in detail below. At channel set-up, the controller 235 stores information in the channel

identifying table 240 that for each time slot identifies the channel to which the time slot is allocated.

The count of the above mentioned slot counter included in the port 225 is used to address an entry in the channel identifying table 240. The channel identifier found at the entry is outputted from the table 240 and is transferred to the mapping means 250 and 251.

For each output bitstream 220 and 221, the mapping means 250, 251 uses the channel identifier received from the channel identifying table 240 to derive a time slot position of bitstream 220 and a time slot position of bitstream 221, for example as will be described below with reference to Fig. 6a and 6b. Using the respective position information to address time slot positions of respective frame buffers 260, 261, the associated data from the input port 225 is written into the frame buffers 260 and 261, and is thus stored at positions given by said mapping means 250, 251.

In similar to port 225, each one of the ports 230 and 231 also includes a bit clock, a slot counter and a frame clock (neither of which is shown in the figure). The count of the slot counter is used to address entries in the frame buffer 260. Based thereupon, data stored in said frame buffer is transmitted as bitstream 220 and 221.

In the channel identifying table 240 in Fig. 2, which is also illustrated more in detail in Fig. 1, time slot positions 1, 2 and 5 in the cycles of bitstream 215 are allocated to the channel denoted A, and time slot positions 3 and 7 are allocated to channel B. Since neither one of time slot positions 4 and 6 in bitstream 215 is allocated to a channel to be switched by the switch 210, the table does not contain any information identifying a channel for the corresponding entries.

When the port 225 reads, for example, time slot number 2 from a cycle of the first bitstream 215, its slot counter value will address the second entry of the

channel identifying table 240. At this entry, information, in this case denoted A, identifying channel A, has been stored by means of the controller 235 during circuit set-up. This channel identifier is thereby associated with time slot number 2, or with the data transferred by time slot number 2. Based upon this channel identifier A, the mapping means 250, 251 will, if so configured by the controller 235, output one of the time slots allocated to channel A in the second bitstream and in the third bitstream. This procedure is repeated continuously for all time slots of the first bitstream, at the start of each new cycle in the first bitstream the slot counter is restarted by the frame clock.

A switch 310 according to a second embodiment of the invention will now be described with reference to Fig. 3. The switch 310 includes a first port 325 arranged to receive a first bitstream 315 and a second and a third port 330 and 331 arranged to transmit a second and a third bitstream 320 and 321, respectively.

In this embodiment, the switch 310 comprises a medium on which an internal bitstream 370 propagates. Each time slot received by the port 325 has a corresponding entry in a slot mapping table 338, and each entry thereof that represents a time slot carrying data to be switched by the switch 310 designates a time slot position of the internal bitstream 370.

When, for example, the first port 325 reads time slot position 5 from a cycle of bitstream 315, its slot counter value will address the fifth entry of the slot mapping table 338. At this entry, position information, in this case for example position number 4, has been stored by means of a controller (not shown) during channel set-up. Upon addressing the fifth entry, the position number 4 will be outputted and used to control that the data read from time slot number 5 of the cycle of the first bitstream is written into time slot number 4 of the cycle of the internal bitstream.

In connection with the second and third port, the time slots of the internal bitstream are read. Upon reading, for example the above mentioned time slot number 4 of the internal bitstream, a pointer will address the second entry of each one of two channel identifying tables 340, 341. At this entry, information, in this case denoted A, identifying the channel A has been stored by means of the contreller during channel set-up. This information is thereby directly associated with time slot number 4 of the internal bitstream 370, and indirectly, via the slot mapping table 338, associated with time slot number five of the first bitstream 315, i.e. indirectly with the data transferred by time slot number 4 of the first bitstream 315.

When reading the fourth time slot position of the internal bitstream 370, i.e. when addressing the fourth entry of the channe identifying tables 340 and 341, information identifying a channel, will be outputted, and based upon such channel identification, the mapping means 350, 351 output a time slot position allocated to said channel A in the second bitstream 320 and a time slot position allocated to said channel on the third bitstream 321. The time slot positions are used to address corresponding entries in frame buffer 360, 361, at which entries data read from the internal bitstream will be stored.

As is understood, in the channel identifying tables 340, 341, which for example could be similar to the table 240 described with reference to Figs. 1 and 2, each entry corresponds to a position of a time slot in the cycles of the internal bitstream 370 and provides information as to whether or not the associated time slot is part of a channel to be transmitted on the respective output bitstream.

In Fig. 4, a switch 410 according to a third embodiment of the invention is shown. In similar to the embodiments above, the switch 410 is connected via ports 425,

430, and 431 to respective bitstreams 415, 420 and 421. The switch 410 further includes means for connecting data read from bitstream 415 with a channel identifier header, these connecting means having the reference numeral 442.

Each time slot in the first bitstream read by the port 325 has a corresponding entry in the channel identifying table 440, and each entry that represents a time slot carrying data to be switched by the switch provides information for identifying a channel, in similar to the table 240 described with reference to Figs. 1 and 2.

When port 425 reads, for example, time slot number 5 from a cycle of the first bitstream 415, its slot counter value will address the fifth entry of the channel identifying table 440. At this entry, information identifying a channel, for example channel A, has been stored by means of the controller (not shown) during channel set-up. Upon addressing said entry of the channel identifying table, the channel identifier A will be outputted and transferred to the connecting means 442. For each time slot read by port 425, data is transferred from the time slot of the first bitstream to the connecting means 442.

Hence, in this example, data read from time slot number 5 of bitstream 425 is transferred to the connecting means 442 at the same time as the associated channel identifier is derived by the channel identifying table 440. The connecting means 442 will then connect said channel identifier with said data. The channel identifier will then at a later step be used by mapping means 450, 451 for data output. The mapping means will use the channel identirier to derive a time slot position in each respective output bitstream and will use this position to address a corresponding entry in respective frame buffer 460, 461, at which entry the data connected to said channel identifier.

Fig. 5 shows a switch according to a fourth embodiment of the invention, which in many regards are similar to the embodiment described above with reference to

Fig.3. However, whereas data read by the input port in Fig. 3 is mapped into a time slot position of an internal bitstream, and a channel identifier is associated with said time slot position of the internal bitstream, data is in Fig. 5 insatead written into a memory location of a shared memory 542. Channel identifying means 540 and 541 is then arranged to associate memory positions of the shared memory 542 with channel identifiers, thereby indirectly associateing the data being stored at said memory locations with said channel identifiers.

For each time slot to be transmitted by output ports 530, 531, respective mapping means 550, 551 stores information as to which channel that the respective output slot refer to. The mapping means will use the output time slot position to address a respective frame buffer 560, 561, and will at the same time use the associated channel identifier to address the respective channel identifying means 540, 541. Based upon the channel identifier, the channel identifying means will in turn produce a memory address to the shared memory, thereby causing readout of data from said memory location to the respective output frame buffer 560, 561, and to the position thereof indicated by the time slot position stated by the respective mapping means 550, 551.

Exemplifying embodiments of mapping means included in the switches shown in Figs. 2-5, for example the mapping means 250, 251, 350, 351, 450, 451, 550, or 551, will now be described with reference to Figs. 6a and 6b. The mapping means 650 basically includes two tables, a channel-to-slot table 640 and a slot-to-next slot table 660. At channel set-up, the above mentioned controller stores information in the channel-to-slot table and in the slot-to-next slot table. The channel-to-slot table has entries that are addressed by a channel identifier. Each entry of the channel-to-slot table 640 contains a time slot position referring to an entry of the slot-to-next slot table 660. In turn, each entry of the slot-to-

next slot table contains a time slot position number of the respective output bitstream, i.e. the time slot position corresponding to the position of the next time slot of the channel in a round-robin fashion. The content of the slot to next slot table 660 is, upon addressing of the corresponding entry, written back to the entry currently being addressed in the channel-to-slot table 640.

The output of the mapping means 650 is used to address the frame buffer, as described above. The time slot position output is either outputted to the frame buffer from an entry of the channel-to-slot table 640, as in the embodiment shown in Fig. 6a, or from an entry of the slot-to-next slot table 660, as in the embodiment 651 shown in Fig. 6b.

As an example, assume that the time slots positions 1 and 5 of the output bitstream have been allocated to a channel B. The information written by the controller in the tables in accordance with these allocations are shown in the figures 6a and 6b. The first time the channel-to-slot table 640 is addressed by a channel identifier B, the entry corresponding to this identifier will output position data that addresses the fifth entry of the slot-to-next slot table 660. In turn, the time slot position indicated at this fifth entry of the slot-to-next-slot mapping table 660 (i.e. identifying slot position 1) will be written back to the channel-to-slot table 640 at the entry given by channel identifier A. The data at this fifth entry is however also used for addressing the frame buffer. The next time the channel-to-slot table 640 is addressed by a channel identifier denoted B, the same entry of the channel-to-slot table will be addressed, but the entry will point to the first entry of the slot-to-next slot table, and so on. This process is repeated continuously, providing an indefinite linked list of time slot positions 1, 5, 1, 5..., and so on.

Referring now to Fig. 7, a principle structure of yet another embodiment of mapping means according to the

present invention will be described. The mapping means 750 includes a set of channel specific FIFO buffers 780, a FIFO buffer selection means 770 and a slot-to-channel mapping table 790. Each FIFO buffer in the set of channel specific FIFO buffers corresponds to a respective channel and each buffer temporarily stores data intended for a respective channel.

Upon receiving a channel identifier, the FIFO buffer selection means 770 selects in which FIFO buffer, out of the set of FIFO buffers, the time slot data associated with said channel identifier is to be stored. This is accomplished by enabling the particular FIFO buffer to accept said time slot data presented to it.

The slot-to-channel mapping table 790 has entries that are cyclically and sequentially addressed, preferably as generated by a slot counter of the output port. Each entry of said table contains channel identifiers, or FIFO buffer identifiers, used for enabling the reading of a specific buffer as indicated by said identifier. As a result of addressing the slot-to-channel mapping table with a certain time slot position number, data will be read from the FIFO buffer storing data belonging to the channel to which the time slot with the corresponding position in the output bitstream is allocated to. This data is then outputted from the mapping means. However, if the FIFO buffer in question is empty, the mapping means will output idle data. The data, or idle data, is then transferred directly transmitted as an output bitstream or is temporaily stored in an output frame buffer.

As is understood, the description above of exemplifying embodiments of the invention has been made in order to provide a better understanding thereof. Of course, an actual switch will incorporate elements not shown in the figures, and may also be realized using other components than the ones specifically described herein. For example, at different locations in the switch, further frame

buffers, multiplexing means, and the like, may be provided to facilitate the desired operation.

Whereas the embodiments described above discusses the situation wherein a channel is switched from an input bitstream to two or more output bitstreams, this does not mean that other channels may simultaneously be switched to only one output port or merely terminate at the switch and hence not be switched at all.

As is understood by those skilled in the art, even though the invention has been described with reference to exemplifying embodiments thereof, different alterations and combinations may be made thereof within the scope of the invention, which is defined by the accompanying claims.